**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

**ЗАДАЧА О ЧИТАТЕЛЯХ И ПИСАТЕЛЯХ-2 («ГРЯЗНОЕ ЧТЕНИЕ»).**

**Пояснительная записка**

Дисциплина: «Архитектура вычислительных систем»

Исполнитель:  
студент группы БПИ198  
Гудзикевич М. С.

**Москва 2020**

СОДЕРЖАНИЕ

[1. ТЕКСТ ЗАДАНИЯ 3](#_Toc58447265)

[2. ПРИМЕНЯЕМЫЕ РАСЧЕТНЫЕ МЕТОДЫ 3](#_Toc58447266)

[3. ТЕСТИРОВАНИЕ ПРОГРАММЫ 4](#_Toc58447267)

[ИСТОЧНИКИ 5](#_Toc58447268)

1. ТЕКСТ ЗАДАНИЯ

Задача о читателях и писателях-2 («грязное чтение»). Базу данных разделяют два типа потоков – читатели и писатели. Читатели выполняют транзакции, которые просматривают записи базы данных, транзакции писателей и просматривают, и изменяют записи. Предполагается, что в начале БД находится в непротиворечивом состоянии (т. е. отношения между данными имеют смысл). Транзакции выполняются в режиме «грязного чтения», то есть процесс-писатель не может получить доступ к БД только в том случае, если ее занял другой процесс-писатель, а процессы-читатели ему не мешают. Создать многопоточное приложение с потоками-писателями и потоками-читателями. Реализовать решение, используя семафоры, и не используя блокировки чтения-записи.

1. ПРИМЕНЯЕМЫЕ РАСЧЕТНЫЕ МЕТОДЫ

Программа работает по следующему алгоритму: считывает данные из аргументов консоли (это четыре целых числа через пробел от 1 до 100 – количество читателей, писателей, размер базы данных и максимальное количество изменений базы данных). Отдельно стоит заметить, что при запуске через консоль могут возникнуть проблемы – если не прописаны пути до нужных библиотек. Далее программа проверяет корректность параметров и, если есть ошибка, – приходит уведомление. Иначе создаётся база данных в непротиворечивом состоянии (вектор упорядоченных по возрастанию целых чисел от 0 до [размер – 1]) и необходимое количество потоков-читателей и писателей (с помощью библиотеки pthread и метода pthread\_create). Потоки работают следующим образом:

Поток-читатель раз в случайно генерируемое время (для замирания используется метод Sleep() из windows.h) генерирует случайный индекс и считывает элемент базы данных с этим индексом.

Поток-писатель, если сейчас ни один из писателей не редактирует базу данных (за это отвечает семафор semaphore из semaphore.h) и количество проделанных изменений не превышает максимума, генерирует случайный элемент, затем вставляет его в БД так, чтобы она была не противоречивой (для этого используется метод бинарного поиска), иначе поток ждёт освобождения семафора (в котором всегда одно вакантное место, так как по условию всего один писатель может вести работу).

По окончании работы выводится измененная база данных, также, в ходе выполнения, все действия логируются.

1. ТЕСТИРОВАНИЕ ПРОГРАММЫ

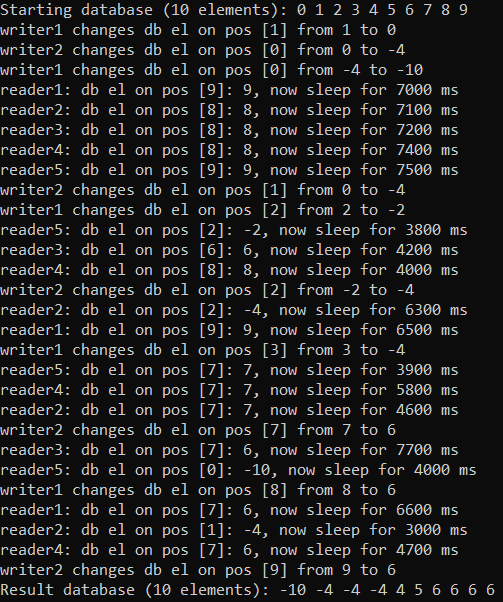
**

Рис. 1 – «Длинный тест» - 5 читателей, 2 писателя, 10 – размер БД, макс. 10 изменений

**

Рис. 2 – «Маленький тест» - 1 читатель, 1 писатель, 1 – размер, 1 итерация



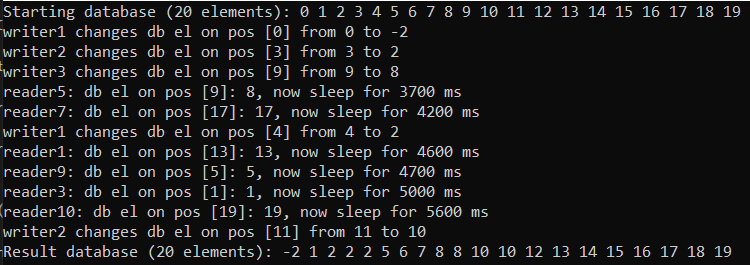
Рис. 3, 4 – сообщения об ошибке при аргументах {1 2 3} и {101 2 3 4} соотв.

Рис. 5 – «Широкий тест» - 10 читателей, 3 писателя, 20 – размер БД и 5 изменений.

ИСТОЧНИКИ

1. SoftCraft, сайт по учебной дисциплине. [Электронный ресурс] http://softcraft.ru/ (дата обращения: 08.12.2020).
2. Cppreference, сайт по C++. [Электронный ресурс] <https://en.cppreference.com/w/cpp/thread/counting_semaphore> (дата обращения: 09.12.2020).
3. Wikipedia. [Электронный ресурс] <https://ru.wikipedia.org/wiki/Задача_о_читателях-писателях> (дата обращения: 09.12.2020).
4. Записки программиста. [Электронный ресурс] <https://eax.me/pthreads/> (дата обращения: 09.12.2020).